Prior Art

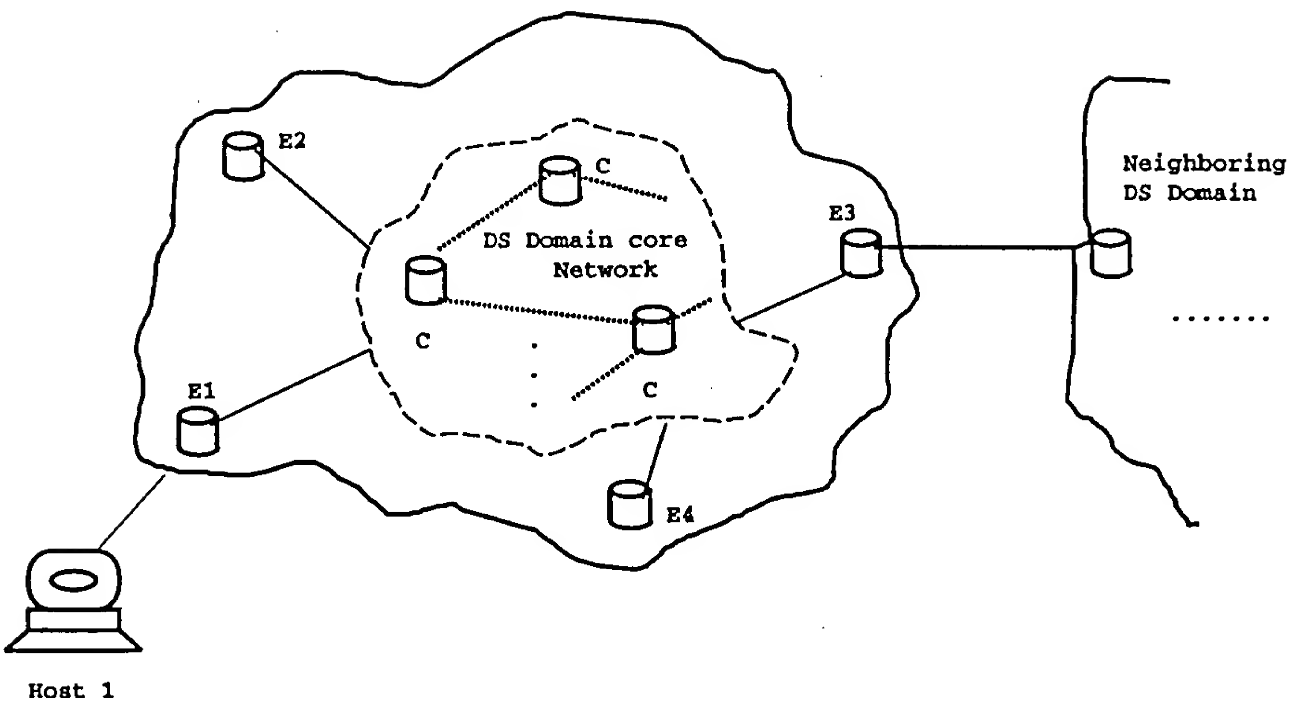


Fig. 1

DIFF-SERV DOMAIN

On every packet arrival

Calculate the average queue size based on exponential moving weighted average 100
if (average queue size < $minth$) enqueue the packet 110

if (minth < average queue size < FeedbackThreshold)
{

enqueue the packet, 120
mark the bits (bit1,bit2) for all outgoing packets queue with (1,0), if the bits are not previously set as (1,1) 130
}
if ($FeedbackThreshold \leq$ average queue size < $maxth$)
{

drop or enqueue the packet with the probability as decided by RED 140
mark the bits (bit1,bit2) for all outgoing packets with (1,1) 150

}

if (average queue size > $maxth$) drop the incoming packets 160

Fig. 2
MODIFICATIONS TO THE RED ALGORITHM AT CORE NODES

Fig. 3

A SIMPLE TWO-BIT SCHEME FOR REPRESENTING LOCAL DOMAIN CONGESTION.

Bit1	Bit2	Inference at the egress node
0	0	No congestion detected so far up to this domain
0	1	No local congestion, but Congestion occurred in a prior domain
1	0	Local congestion occurred, but no packet loss phase
1	1	Local congestion occurred and in packet loss phase

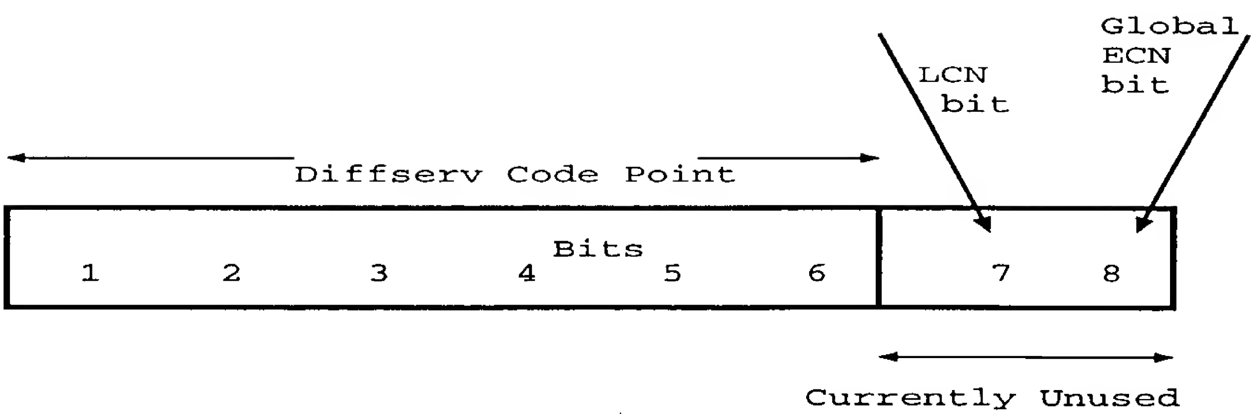
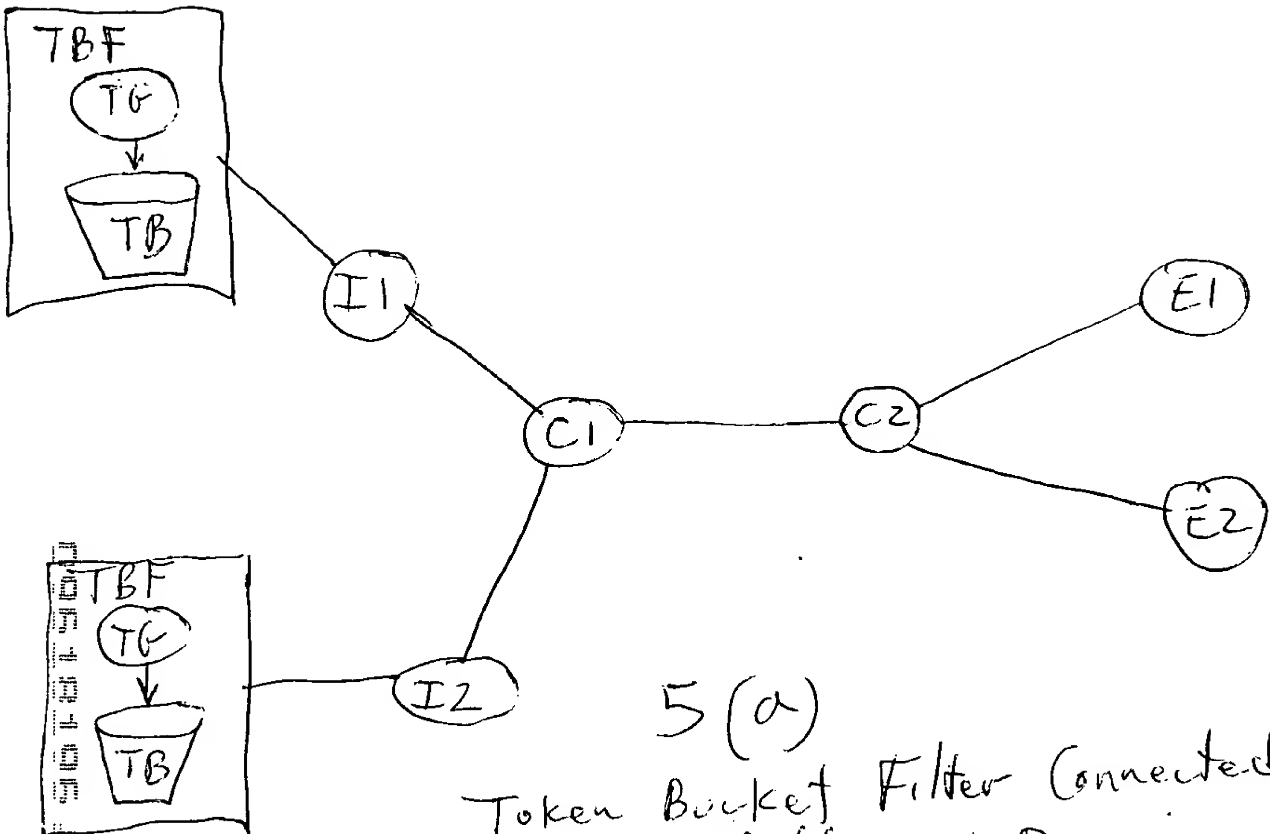
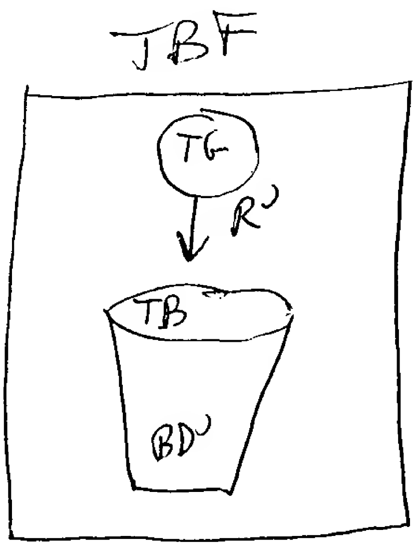


Fig. 4

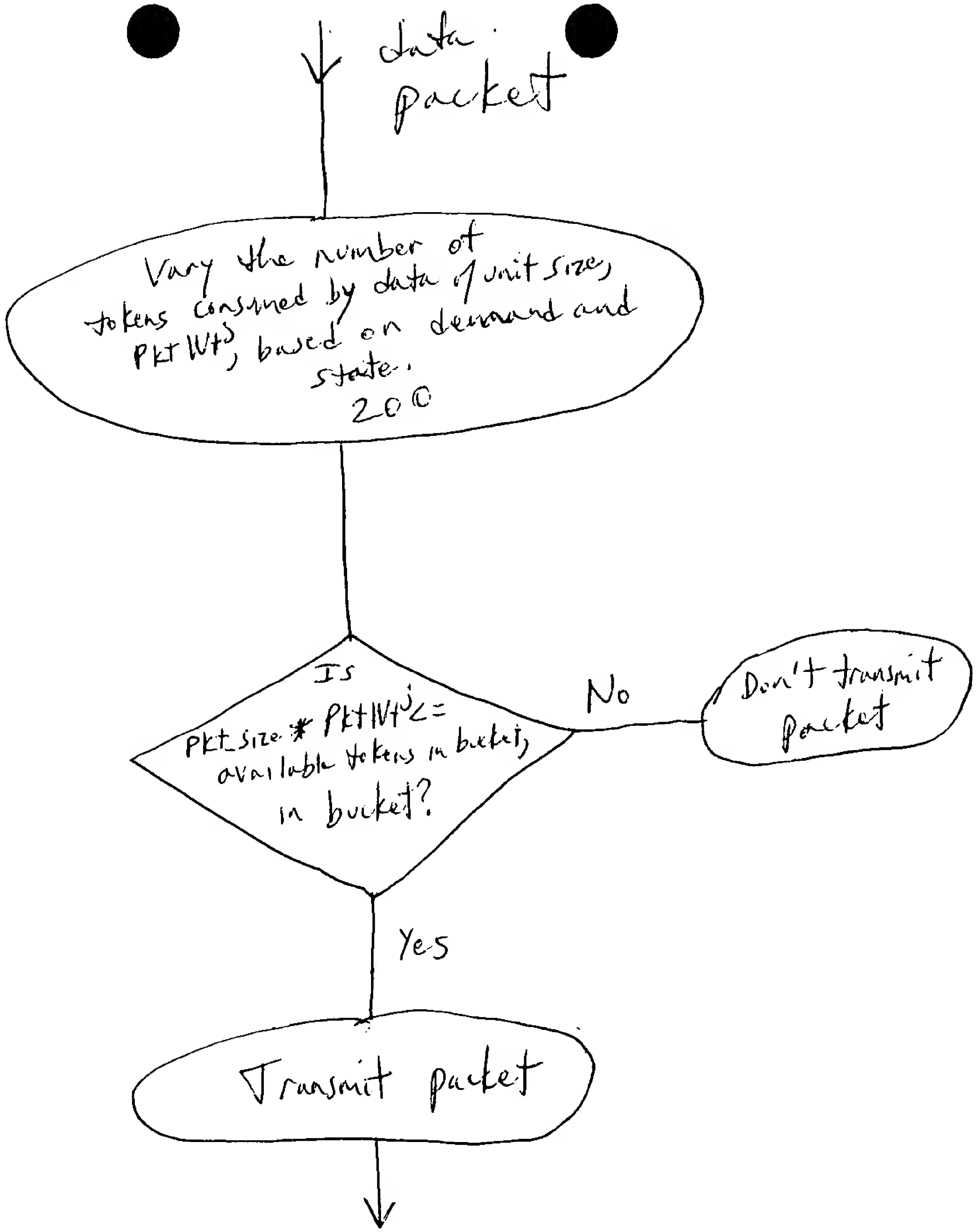
THE TOS/DSCP BYTE



5(a)
Token Bucket Filter Connected to Diff-serv Domain



5(b)
Components of a Token Bucket Filter



Flowchart for TBF-Based Rate Control Method

Fig. 6a

Initialize: $PktWt_0^j \leftarrow 1.0$
$PktWt^j$ is always within $[minPktWt^j, maxPktWt^j]$
MD is a monotonously decreasing function that takes a value (0,1]
MI is a monotonously increasing function that takes a positive value
j denotes the label corresponding to fixed route between a given pair of ingress/egress nodes

for every ith round trip time (between ingress and egress nodes)

during congestion-free periods

if(average TBF queue size at ingress node $\geq DemandThrsh^j$)

$PktWt_i^j \leftarrow PktWt_{i-1}^j * MD(PktWt_{i-1}^j)$ 230

/* decrease the $PktWt^j$ during congestion free periods, based on demand at TBF */

else {
if ($PktWt_{i-1}^j > 1$) $PktWt_i^j \leftarrow max[1, PktWt_{i-1}^j * MD(PktWt_{i-1}^j)]$

if ($PktWt_{i-1}^j < 1$) $PktWt_i^j \leftarrow min[1, PktWt_{i-1}^j * MI(PktWt_{i-1}^j)]$ }
/* restore $PktWt^j$ close to 1.0 */

At congestion notification time

$PktWt_i^j \leftarrow \frac{(maxPktwt^j - 1)(1 - Pktwt_{i-1}^j)}{(1 - minPktWt^j)} + 1$ if $PktWt_{i-1}^j < 1$.

/* The smaller the $PktWt^j$ just before LCN, the bigger it will be during congestion period. A uniform mapping of $[minPktWt^j, 1)$ on to $(1, maxPktWt^j]$ intervals */ 250

During congestion period

$PktWt_i^j \leftarrow PktWt_{i-1}^j * MI(PktWt_{i-1}^j)$ if $PktWt_{i-1}^j \neq 1$ 240

On receipt of congestion clearance notification

Select a random time less than RTT and,

$PktWt_i^j \leftarrow PktWt_{i-1}^j * MD(PktWt_{i-1}^j)$ 220

Fig. 6(b)
THE TBF-BASED CONGESTION MANAGEMENT ALGORITHM AT INGRESS NODES

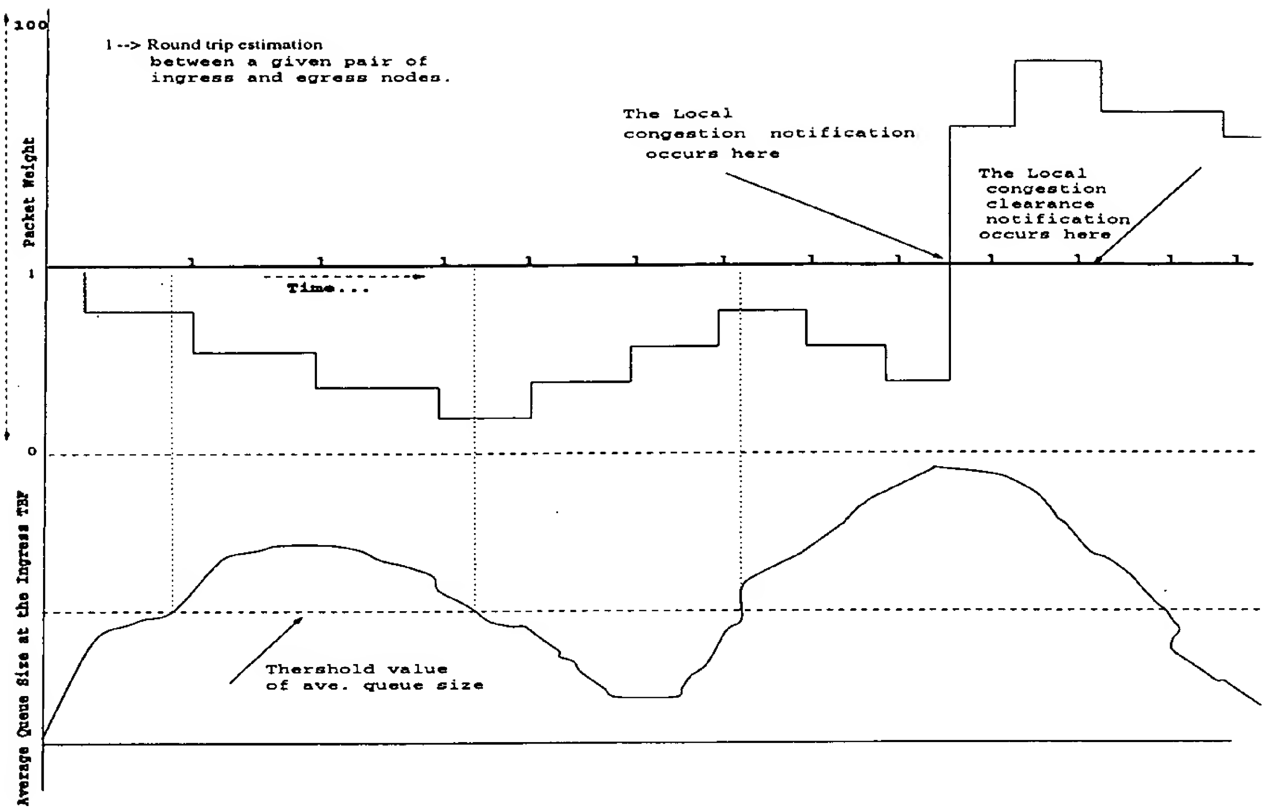


Fig. 7

VARYING OF PKT_WT WITH DEMAND AND LCN MESSAGES

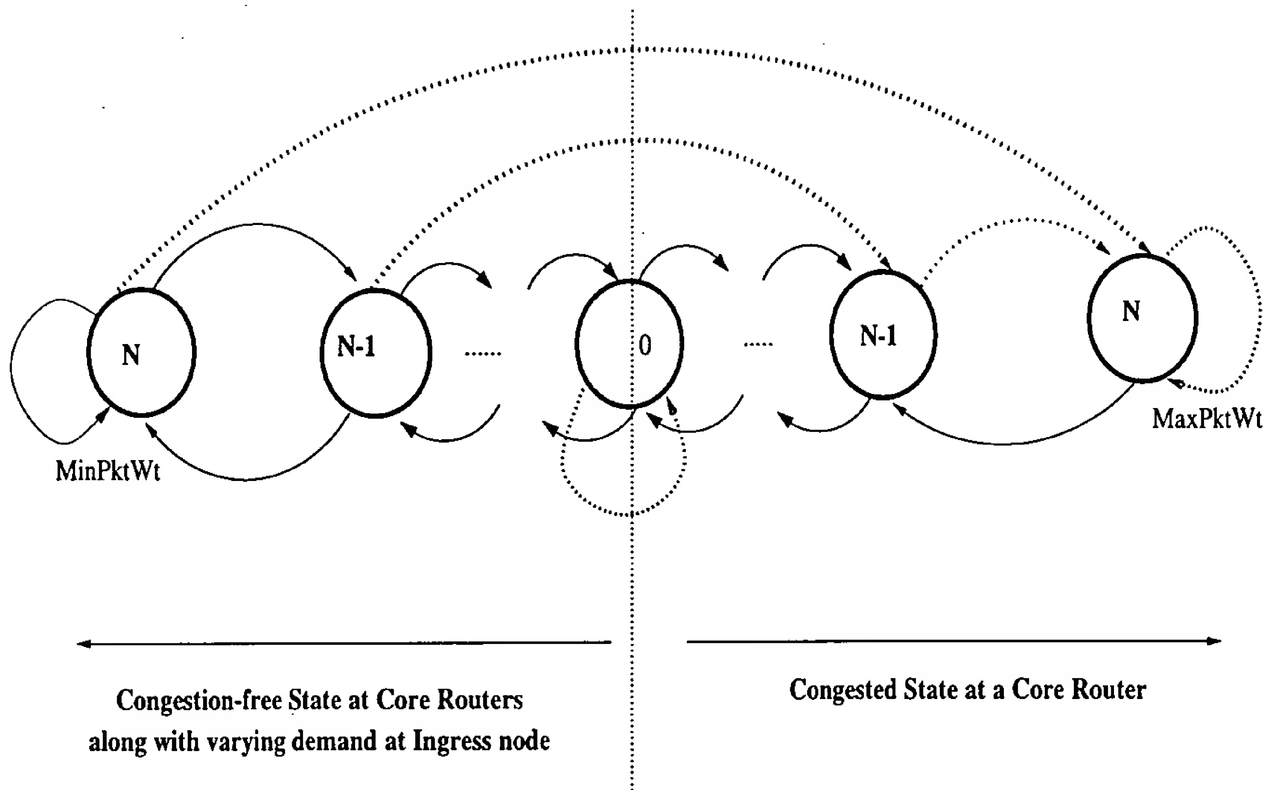


Fig. 8

STATE DIAGRAM OF PKTWT DYNAMICS

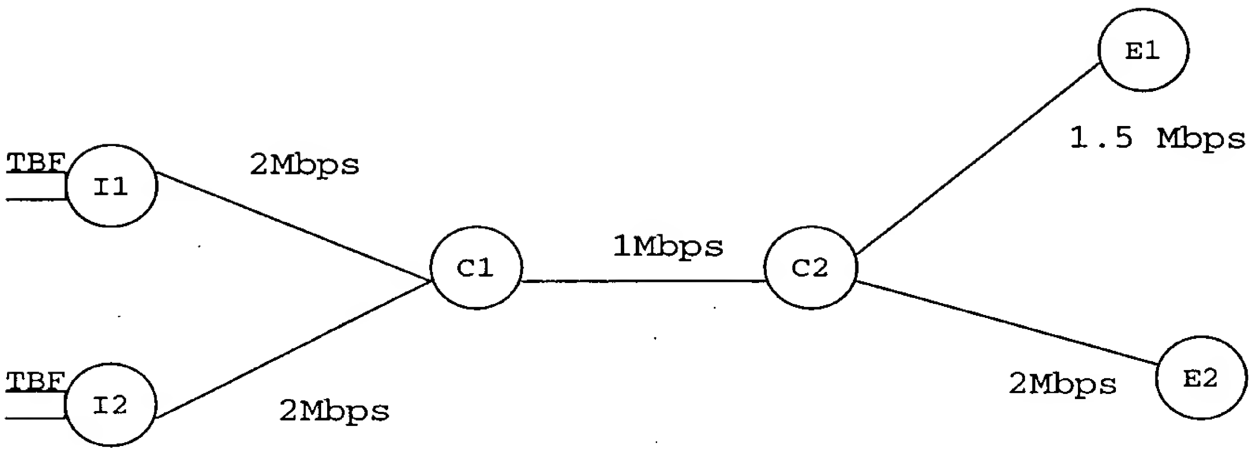


Fig. 9

THE SIMULATION SETUP

Fig. 10

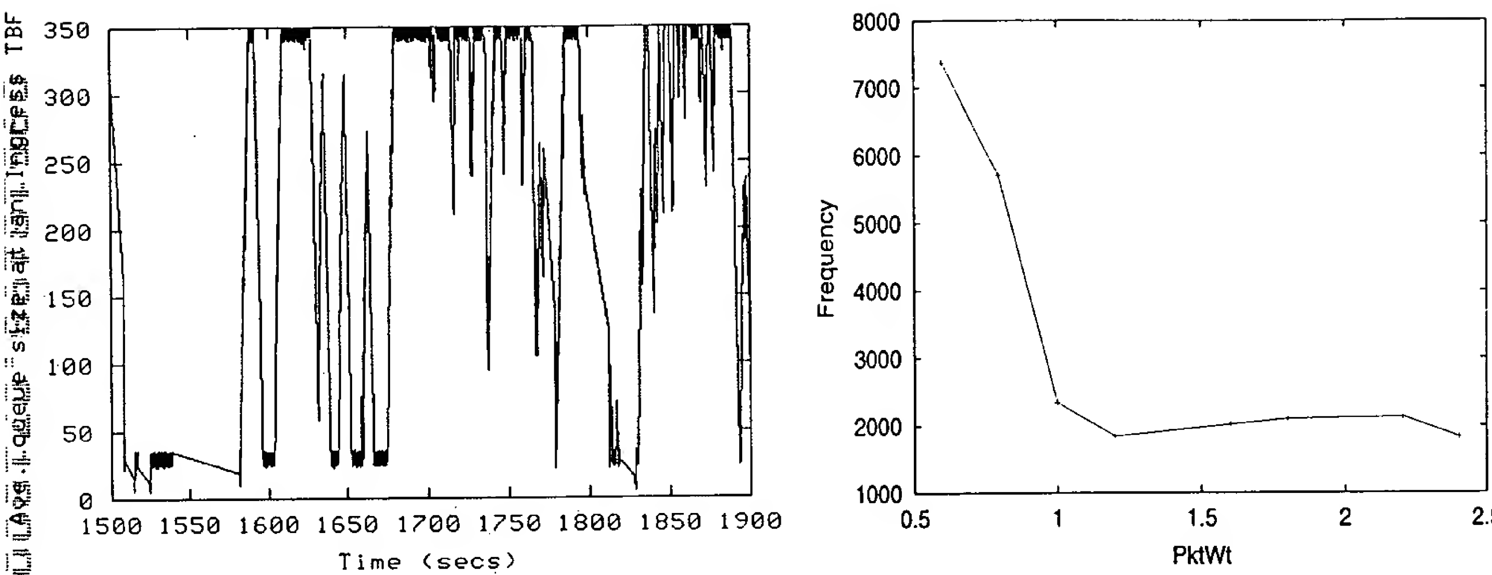
PERFORMANCE OF THE PROPOSED DCM SCHEME

	Non feedback scheme RED (core), % pkt loss	DCM scheme % of packet loss			
Utilization	at core nodes (x)	at core nodes	at ingress TBFs	overall loss core+TBFs (y)	Overall improvement with DCM (relative reduction $\frac{(x-y)}{x}$ in pkt loss)
0.5	3.88	1.0859	1.4889	2.5748	33.76
0.6	8.00	2.1948	2.7775	4.9723	37.87
0.7	11.4	2.8461	3.9036	6.7498	40.79
0.8	12.8	2.8148	4.5384	7.3531	42.55
0.9	14.1	2.7192	6.3322	9.0514	35.81
1.0	16.6	2.6678	7.6945	10.3623	37.59
1.1	18.3	2.9650	10.3028	13.2677	27.54
1.2	19.3	2.8883	11.4976	14.3858	25.49
1.3	20.76	2.8530	12.7693	15.6223	24.75

Fig. 11

DELAY PERFORMANCE OF THE DCM SCHEME

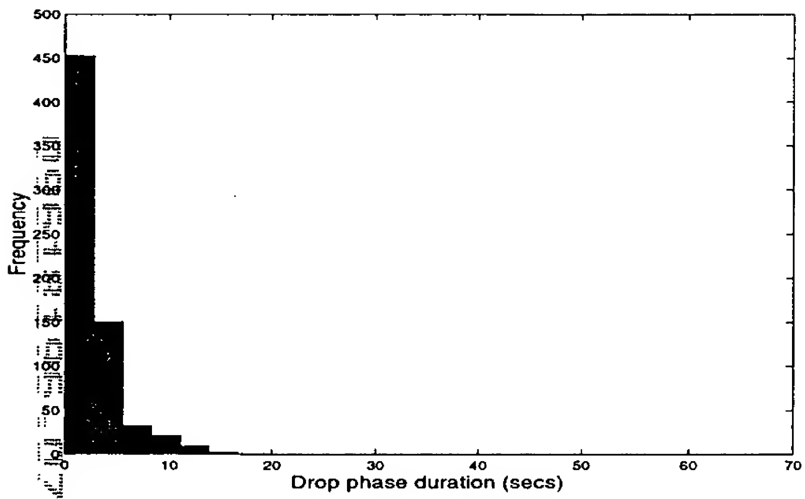
Utilization	Average delay (seconds) at ingress TBFs
0.8	0.771937
0.9	0.924975
1.0	1.007773
1.1	1.273592
1.2	1.339390
1.3	1.389371

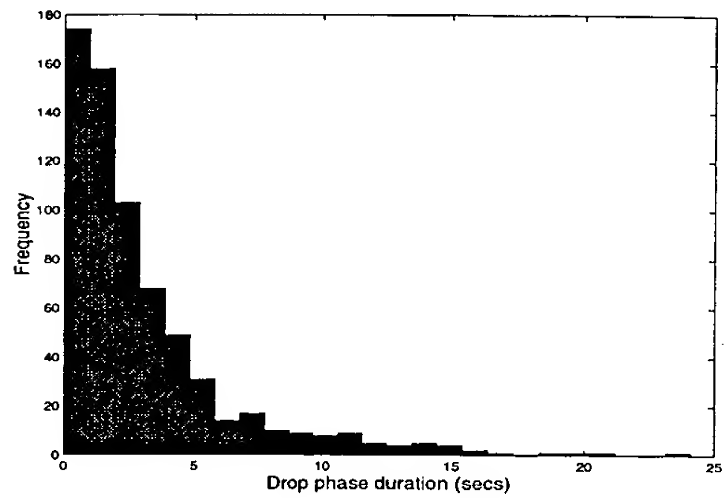


(a) Ave. queue size at an ingress node;

(b) PktWt distribution ; util. = 0.8

Fig. 12



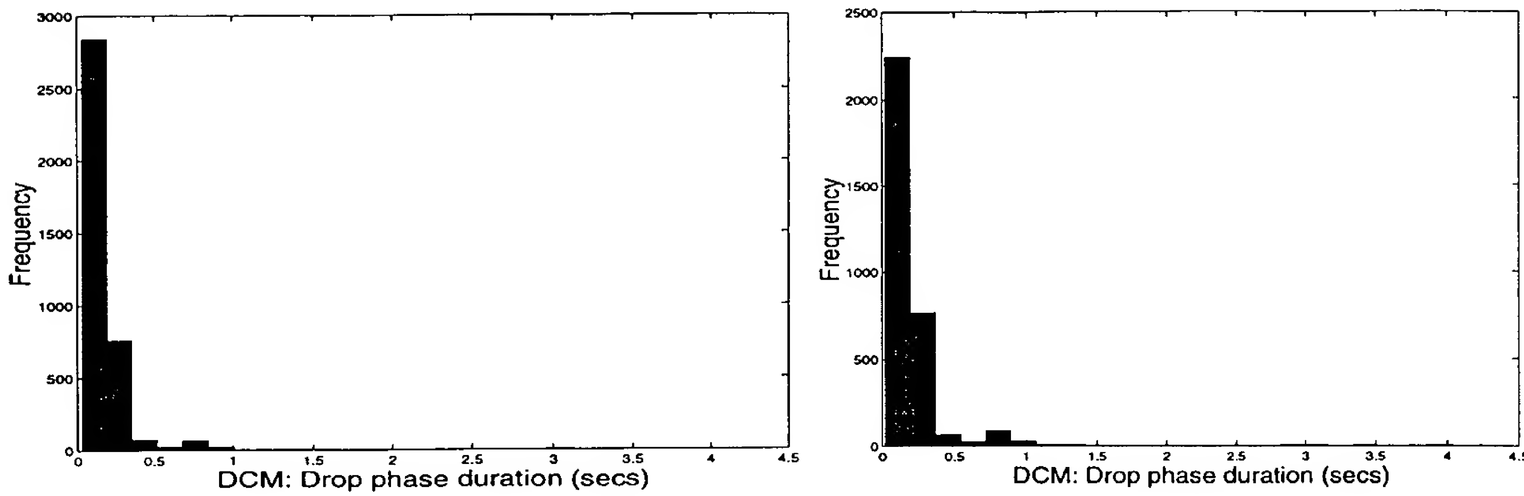


(a)non-DCM scheme at Utilization = 0.8;

(b)non-DCM scheme at Utilization = 0.9

Fig. 13

DISTRIBUTION OF PACKET DROP PHASE DURATION AT THE CORE NODES WITH NON-DCM SCHEME

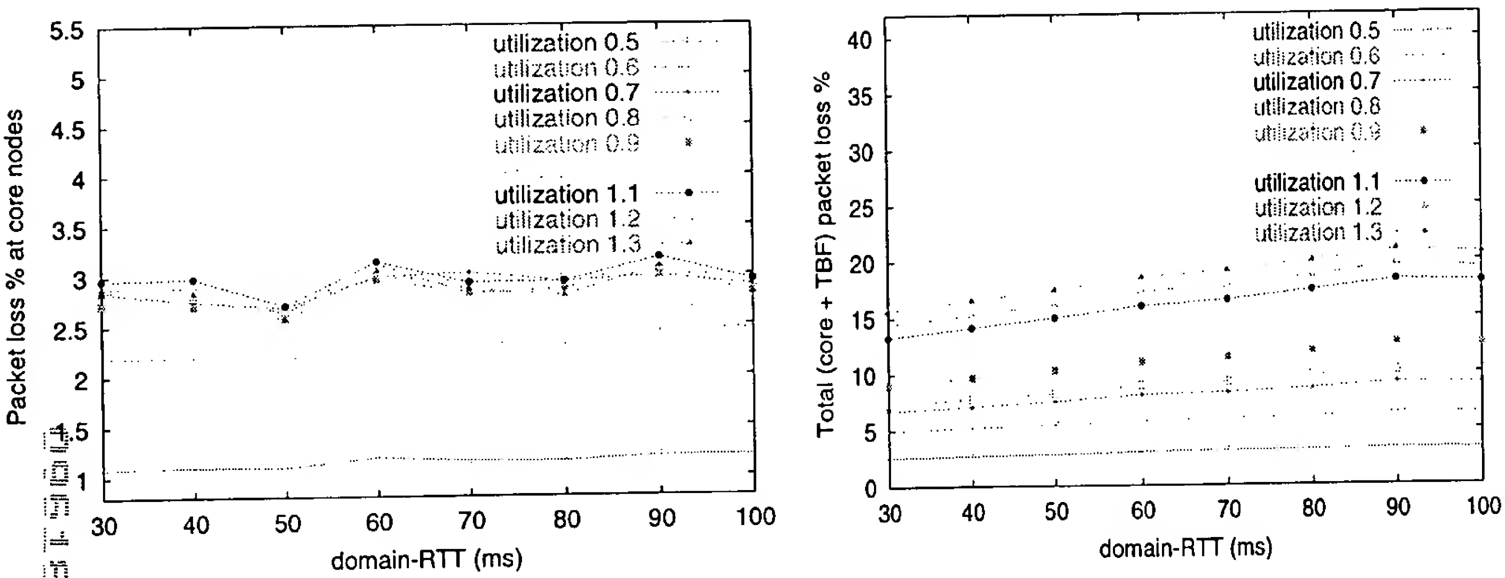


(a) DCM scheme Utilization = 0.8;

(b)DCM scheme at Utilization = 0.9

Fig. 14

DISTRIBUTION OF PACKET DROP PHASE DURATION AT THE CORE NODES WITH DCM SCHEME



(a) packet loss % at core nodes;

(b) Total packet loss in the system (core +TBF)

Fig. 15

PERFORMANCE OF THE DCM SCHEME WITH DOMAIN-RTT VARIATION